Linac Server Node

Problem analysis

Mon, Dec 16, 2002

Recently the Linac server node, `node0600`, was observed to fail in a way that was not seen before. In response to `RETDAT` protocol requests, an error code was returned that indicated that the `LISTP` table was full. This note describes the analysis.

The `LISTP` table is used to house request block pointers for active requests. The size of this table in `node0600` permits up to 240 entries to be active. There is a header field that shows how many data requests are active. If the `LISTP` table was full, then it was thought that this number might show this problem. So we defined a device that can hold this value and added it to the Acnet data logging system. We also plotted it occasionally on the Macintosh Parameter Page. Its normal value is about 20 and was never seen to rise as high as 40.

From Kevin Cahill we learned that Bob West had been doing some accessing of Acnet device data using whatever devices he found in the Acnet database. But he was on vacation last week, and no repeat of the previous week's failures occurred. Upon his return we did some testing, monitoring the count of active requests while he performed the same work as he did earlier. There was no evidence of a problem; the counter typically had values of about 25. He kept his test program running for awhile just to see whether any problem showed up later.

After some time, we learned that there were errors occurring that included the same error code; the `LISTP` table was full. Examining that table showed that indeed it was full, even if the counter did not indicate as much.

Although the counter is in the header of the `LISTP` table, it does not actually refer directly to the number of used entries in that table. It actually shows the number of requests that are active, of any type, in the system. This count is determined by the code that traverses the linked list of active requests every 15 Hz cycle, in order to fulfill all those for which replies are due on the present cycle.

So, why does the count not match the number of used entries in the table? Examining the `RETDAT` code, the `NewListN` function is called every time a server requests needs to obtain a list number, and a pointer to an allocated request block is placed into the new entry. When the request is later freed, the request block is freed, and the `LISTP` entry is also freed.

But there is an occasion when this does not happen. If a `RETDAT` request is being initialized, and a `LISTP` entry has been allocated, but an error is subsequently detected, the request is rejected, which results in the request block being freed. Unfortunately, for this case, the `LISTP` entry is not freed. This is a bug that should be easy to fix. Analogous logic exists for support of an Acnet setting, in which a request-like block is allocated to help return the setting reply status. If an error is detected for such a setting, the block is freed and the `LISTP` entry is also freed. The same needs to be done for a server-style `RETDAT` request in which an error is encountered after the request block and `LISTP` entry have been allocated.